

What is claimed is:

1. A method of exchanging information among applications, the method comprising:
  - providing a plurality of transformers, each transformer corresponding to a unique transformation from one format into another;
  - using a first transformer to transform a data object from a format understandable by a first application into a common format data object;
  - publishing the common format data object to a communication channel;
  - subscribing to the communication channel to retrieve the published common format data object; and
  - using a second transformer to transform the common format data object into a format understandable by a second application.
2. The method of claim 1 wherein the data object corresponds to one or more of a plurality of business events.
3. The method of claim 1 wherein using the first transformer to transform the data object from the format understandable by the first application into the common format data object comprises translating the data object from a vendor-specific format

associated with the first application to an Interface Data Language (IDL) object and storing the IDL object in a shared object model.

4. The method of claim 3 wherein the shared object model comprises a central repository of data objects corresponding to business events.

5. The method of claim 1 wherein using a first transformer to transform the data object from the format understandable by the first application into the common format data object is performed in response to a recognition of a business event by the first application.

6. The method of claim 1 wherein the method is performed in accordance with a plurality of process models that collectively define when information is to be exchanged among applications.

7. The method of claim 1 wherein publishing the common format data object to a communication channel is performed by a source connector and subscribing to the communication channel is performed by a target connector.

8. The method of claim 1 wherein publishing the common format data object to a communication channel is performed in accordance with a channel architecture that defines a plurality of communication channels having relative priorities.

9. The method of claim 1 wherein using the second transformer to transform the common format data object into the format understandable by the second application comprises retrieving a stored Interface Data Language (IDL) format object from a central repository and translating the IDL object into a vendor-specific format associated with the second application.

10. The method of claim 1 in which information is exchanged among business support systems or operational support systems or a combination thereof.

11. The method of claim 1 in which at least one of the transformers comprises a class defined in an object-oriented programming language.

12. The method of claim 1 further comprising providing, for each transformer, a controller that is configured to route data objects to an associated transformer.

13. The method of claim 12 comprising routing a data object to the first transformer using a first controller.

14. The method of claim 12 comprising routing the common format data object to the second transformer using a second controller.

15. The method of claim 12 in which at least one of the controllers comprises a class defined in an object-oriented programming language.

16. The method of claim 1 further comprising using an acknowledgement class to exchange status messages among applications.

17. The method of claim 16 further comprising using the acknowledgement class to perform exception handling.

18. A method of facilitating the exchange of information among applications, the method comprising:

receiving a data object from a first application;

using a first controller to route the received data object to a first transformer;

using the first transformer to transform the data object from a first format used by the first application into a common format object;

publishing the common format object to a communication channel;

receiving a request from a subscribing application to subscribe to the communication channel;

using a second controller to route the common format object to a second transformer;

using the second transformer to transform the common format object into a data object in a second format used by the subscribing application; and

sending the data object in the second format to the subscribing application.

19. The method of claim 18 wherein the data object received from the first application corresponds to one or more of a plurality of business events.

20. The method of claim 18 wherein using the first transformer to transform the data object from the format used by the first application into the common format object comprises translating the data object from a vendor-specific format associated with the first application to an Interface Data Language (IDL) object and storing the IDL object in a shared object model.

21. The method of claim 20 wherein the shared object model comprises a central repository of data objects corresponding to business events.

22. The method of claim 18 wherein using the first transformer to transform the data object from the format used by the first application into the common format object is performed in response to a recognition of a business event by the first application.

23. The method of claim 18 wherein the method is performed in accordance with a plurality of process models that collectively define when information is to be exchanged among applications.

24. The method of claim 18 wherein, if requests are received from a plurality of subscribing applications, then, for each subscribing application, the common format object is transformed using an associated transformer into a format corresponding to the subscribing application and sent to the subscribing application.

25. The method of claim 18 wherein publishing the common format data object to a communication channel is performed in accordance with a channel architecture that defines a plurality of communication channels having relative priorities.

26. The method of claim 18 wherein using the second transformer to transform the common format object into a data object in the second format used by the subscribing application comprises retrieving a stored Interface Data Language (IDL) format object from a central repository and translating the IDL object into a vendor-specific format associated with the subscribing application.

27. The method of claim 18 in which information is exchanged among business support systems or operational support systems or a combination thereof.

28. A system for facilitating the exchange of information among applications, the system comprising:

a plurality of process models each defining one or more conditions for sending a business event from an application to one or more other applications;

a shared object model configured to store data objects received from applications in a common format;

a plurality of transformer classes configured to translate data object from a format used by one or more applications into the common format or vice versa; and

a plurality of controller classes configured to route data objects to associated transformer classes.

29. The system of claim 28 further comprising a channel architecture defining a plurality of communication channels to which data objects from an application are to be published.

30. The system of claim 29 wherein the channel architecture defines relative priorities for the plurality of communication channels.

31. The system of claim 28 further comprising an acknowledgement class configured to exchange status messages among applications.

32. The system of claim 31 wherein the acknowledgement class is further configured to perform exception handling.

33. The system of claim 28 wherein each process model corresponds to a different business event.

34. The system of claim 28 wherein the shared object model comprises a central repository of data objects in an Interface Description Language (IDL) format.



35. The system of claim 28 wherein each transformer class corresponds to a unique application format-common format translation.

36. The system of claim 28 wherein each controller class is configured to route data objects to an associated transformer class according to a process model.

37. The system of claim 28 wherein the transformer classes and the controller classes are implemented as classes in an object-oriented programming language.

38. Machine-readable instructions, embodied in a tangible medium or as a propagated signal or both, for facilitating the exchange of information among applications, execution of the instructions causing one or more machines to perform operations comprising:

receiving a data object from a first application;

using a first controller to route the received data object to a first transformer;

using the first transformer to transform the data object from a first format used by the first application into a common format object;

publishing the common format object to a communication channel;

receiving a request from a subscribing application to subscribe to the communication channel;

using a second controller to route the common format object to a second transformer;

using the second transformer to transform the common format object into a data object in a second format used by the subscribing application; and

sending the data object in the second format to the subscribing application.

39. The instructions of claim 38 wherein the machine-readable instructions comprise computer software instructions executable by one or more computer systems.

40. The instructions of claim 38 wherein the data object received from the first application corresponds to one or more of a plurality of business events.

41. The instructions of claim 38 wherein using the first transformer to transform the data object from the format used by the first application into the common format object comprises translating the data object from a vendor-specific format associated with the first application to an Interface Data Language (IDL) object and storing the IDL object in a shared object model.

42. The instructions of claim 41 wherein the shared object model comprises a central repository of data objects corresponding to business events.

43. The instructions of claim 38 wherein using the first transformer to transform the data object from the format used by the first application into the common format object is performed in response to a recognition of a business event by the first application.

44. The instructions of claim 38 wherein one or more of the instructions are executed in accordance with a plurality of process models that collectively define when information is to be exchanged among applications.

45. The instructions of claim 38 wherein, if requests are received from a plurality of subscribing applications, then, for each subscribing application, the common format object is transformed using an associated transformer into a format corresponding to the subscribing application and sent to the subscribing application.

46. The instructions of claim 38 wherein publishing the common format data object to a communication channel is performed in accordance with a channel architecture that defines a plurality of communication channels having relative priorities.

47. The instructions of claim 38 wherein using the second transformer to transform the common format object into the data object in the second format used by the subscribing application comprises retrieving a stored Interface Data Language (IDL) format object from a central repository and translating the IDL object into a vendor-specific format associated with the subscribing application.

48. The instructions of claim 38 in which information is exchanged among business support systems or operational support systems or a combination thereof.